UNITED STATES PATENT AND TRADEMARK OFFICE

Tr

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/762,621	01/22/2004	Andrew Hadley	03-0023	5052

24319 7590 08/04/2006

LSI LOGIC CORPORATION
1621 BARBER LANE
MS: D-106
MILPITAS, CA 95035

EXAMINER
CHU, GABRIEL L

ART UNIT	PAPER NUMBER
2114	

DATE MAILED: 08/04/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 10/762,621	**Applicant(s)** HADLEY ET AL.	
	Examiner Gabriel L. Chu	**Art Unit** 2114	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *22 January 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-43* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-12 and 14-43* is/are rejected.

7)☒ Claim(s) *13* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

Claim Objections

1. Claim 19, 33 objected to because of the following informalities:

Claim 19 appears to have been intended to depend from claim 18 instead of claim 12. Otherwise, "the type and value" does not have antecedent basis. For the purpose of examination, claim 19 will be treated as dependent on claim 18.

Claim 33 refers to "the next parameter" wherein no antecedent has been provided. This is highlighted by claim 40's use of "a next parameter". For the purpose of examination, this "the next parameter" is understood to refer to a parameter following the current parameter in a predefined list of parameters, in the list's order.

Appropriate correction is required.

2. Claim 24 objected to under 37 CFR 1.75(c), as being of improper dependent form for failing to further limit the subject matter of a previous claim. Applicant is required to cancel the claim(s), or amend the claim(s) to place the claim(s) in proper dependent form, or rewrite the claim(s) in independent form. Referring to claim 24, in claim 23, Applicant has already claimed that the states implement the state machine. Further stating that these same states correspond to a state of the state machine does not serve to further limit the subject matter of claims 22, 23, as it has already been claimed the these states implement the state machine.

Claim Rejections - 35 USC § 101

3. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 12-21 rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Referring to claims 12-21, a "computer readable medium having a program of instructions implemented in code" is not viewed to be limited to a computer storage medium storing instructions executed by a computer. Applicant's paragraph 28 of the pre-grant publication further does not limit it to such storage mediums, "e.g., memory 310".

Claim Rejections - 35 USC § 112

4. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

5. **Claims 7, 8 rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement.** The claim(s) contains subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention. Referring to claim 7, 8, Applicant claims "estimating a number of states needed to build the state machine" and "estimating a number of dead states in the state machine at a point in time after the state machine is built". Estimating implies using an algorithm for providing an inexact but approximate value of something. Applicant has provided no description so as to determine the extent or the implementation of such estimating. While it is known that building a state machine requires the use of a number of states, merely stating that an estimate can be made in no way provides a person of ordinary skill in the art with the ability of making an estimate. Further, while

analyzing/processing a specification for use in generating a FSM may provide a determination of a number of dead/states, this does not explain any deviation from such a determination, such as an estimation.

Further regarding claim 8, Applicant has claimed estimating a number of dead states *after* the state machine is built. Examiner notes that Applicant has provided no explanation as to what may constitute a dead state, let alone how to estimate the number of dead states.

6. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

7. **Claim 14 rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.** Referring to claim 14, "PCI cache line" appears to be a misrepresentation of "PCI cache line size" as disclosed in the pre-grant publication paragraph 2.

Claim Rejections - 35 USC § 102

8. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

9. **Claims 1-6, 9-12, 17-19, 21-29 rejected under 35 U.S.C. 102(b) as being anticipated by US 5623499 to Ko et al.** Referring to claim 1, Ko discloses a method for

creating and using a state machine, comprising: creating a first state; creating a second state; building a state machine from the first and second states (Figure 1, Figure 3, 210, 220, Figures 4-8, Figure 10.),

the state machine being capable of performing a plurality of functions (From line 35 of column 3, "The EFSM has a finite number of states and changes from one state to another when an input or stimulus is applied to the machine. A state is defined as a stable condition in which the EFSM rests until the next stimulus or input is applied. Each state transition may also cause the EFSM to update context variables or internal variables, and/or generate observable outputs which may be based on the context variables. The particular final state of an EFSM transition upon receipt of an input or stimulus may be dependent on the current EFSM state, the value of the input parameters, and the current value of context variables or internal variables.");

and using common code for each of the plurality of functions, the common code not being unique for a function of a given type (From the abstract, with emphasis, "**A** method and apparatus for generating a conformance test data sequence of minimal length to verify that a device conforms to a protocol entity which can be characterized by a simplified extended finite state machine. **The** method generates an expanded directed graph of the protocol wherein each state is represented by a state vertex and a dummy vertex which are connected to other similar state and dummy vertices by directed edges in a configuration corresponding to the operation of the machine.").

10. Referring to claim 2, Ko discloses the given type includes at least one of editing, storing, loading, and varying a parameter (From figure 3, 230 retrieve test data set.

From figures 1, 4-8, 10, Ko shows the states receiving inputs and generating outputs of test parameters.).

11. Referring to claim 3, Ko discloses the plurality of functions include editing, storing, loading, and varying a parameter (From figure 3, 230 retrieve test data set. From figures 1, 4-8, 10, Ko shows the states receiving inputs and generating outputs of test parameters. From line 61 of column 11, "According to the test data set the parameter a must be tested with the values 0, 5 and 9 and the parameter b must be tested with the values 1 and 4." From line 61 of column 14, "During a conformance test by the testing system 100 of FIG. 2, the device 110 under test is initialized to state v.sub.1 and the particular sequence of inputs of column 1030 are applied to the device 110. In response, the testing computer 120 receives corresponding generated outputs and compares them with the expected outputs listed in column 1040.").

12. Referring to claim 4, Ko discloses recognizing dead states in the state machine (From figure 5, 420.).

13. Referring to claim 5, Ko discloses removing dead states from the state machine (From line 46 of column 9, "If the starting state v.sub.i as shown by the vertex 410 of an expanded directed subgraph 500 of FIG. 6 is one of the final states of another expanded directed subgraph, such as one of the vertices 420 of the subgraph 400 of FIG. 5, then the subgraphs can be combined to form a larger subgraph of the expanded directed graph. Accordingly, in step (3) of the bte-based direct graph generation method, the respective bte's generated in step (2) are combined to form an expanded directed graph. Applying the bte-based directed graph generation method to the

simplified EFSM 5 of FIG. 1 produces the expanded directed graph 300 shown in FIG. 4.").

14. Referring to claim 6, Ko discloses reviving dead states from the state machine (From line 46 of column 9, "If the starting state v.sub.i as shown by the vertex 410 of an expanded directed subgraph 500 of FIG. 6 is one of the final states of another expanded directed subgraph, such as one of the vertices 420 of the subgraph 400 of FIG. 5, then the subgraphs can be combined to form a larger subgraph of the expanded directed graph. Accordingly, in step (3) of the bte-based direct graph generation method, the respective bte's generated in step (2) are combined to form an expanded directed graph. Applying the bte-based directed graph generation method to the simplified EFSM 5 of FIG. 1 produces the expanded directed graph 300 shown in FIG. 4.").

15. Referring to claims 9, Ko discloses after building the state machine, running the state machine to test a device under test (From the abstract, "A method and apparatus for generating a conformance test data sequence of minimal length to verify that a device conforms to a protocol entity which can be characterized by a simplified extended finite state machine.").

16. Referring to claims 10, Ko discloses after building the state machine, running the state machine to test software under test (From the abstract, "A method and apparatus for generating a conformance test data sequence of minimal length to verify that a device conforms to a protocol entity which can be characterized by a simplified extended finite state machine.").

17. Referring to claims 11, Ko discloses after building the state machine, running the state machine to simulate a device (From the abstract, "A method and apparatus for generating a conformance test data sequence of minimal length to verify that a device conforms to a protocol entity which can be characterized by a simplified extended finite state machine.").

18. Referring to claim 12, Ko discloses a computer readable medium having a program of instructions implemented in code, the program of instructions comprising: creating a first state; creating a second state; building a state machine from the first and second states (Figure 1, Figure 3, 210, 220, Figures 4-8, Figure 10.),

the state machine being capable of executing at least one function (From line 35 of column 3, "The EFSM has a finite number of states and changes from one state to another when an input or stimulus is applied to the machine. A state is defined as a stable condition in which the EFSM rests until the next stimulus or input is applied. Each state transition may also cause the EFSM to update context variables or internal variables, and/or generate observable outputs which may be based on the context variables. The particular final state of an EFSM transition upon receipt of an input or stimulus may be dependent on the current EFSM state, the value of the input parameters, and the current value of context variables or internal variables."),

the at least one function being implemented in code common to multiple parameters (From the abstract, with emphasis, "**A** method and apparatus for generating a conformance test data sequence of minimal length to verify that a device conforms to a protocol entity which can be characterized by a simplified extended finite state machine. **The** method generates an expanded directed graph of the protocol wherein each state is represented by a state vertex and a dummy vertex which are connected to other similar state and dummy vertices by directed edges in a configuration corresponding to the operation of the machine.").

19. Referring to claim 17, Ko discloses a look up table for storing default values of the multiple parameters (From the abstract, "The directed edges are then assigned traversal numbers corresponding to the minimum number of times a respective directed edge need be traversed in order to test values in a predetermined test data set." Table 1.).

20. Referring to claim 18, Ko discloses a look up table for providing type and value information for each of the multiple parameters (From line 6 of column 6, "After the expanded directed graph of the simplified EFSM is generated in step 220, a suitable test data set corresponding to the simplified EFSM is retrieved in step 230. A test data set is a set of input parameters which must be applied, and output parameters which must be evaluated for conformance testing of a device. The particular values of the input parameters and output parameters in the test data set may be specified by designers or conformance testers of the protocol entity. The test data set facilitates the exercise of a state transition several times with different values of the same input and output parameter that may be required to verify that the operation of a device exactly conforms to the protocol entity. For example, consider a transition ##EQU1## which takes an input parameter "phno", changes from a state S.sub.1 to a state S.sub.2, and produces an output parameter "status". Values of the output parameter status are

`invalid`, `local`, `domestic` or `international` depending on the value of the input parameter "phno". The test data set for such a transition may be given in the form of the following input-output table 1." From line 57 of column 11, "A parameter is considered independent from another parameter if it may take on any value in its permitted range without affecting the constraints on the value of the other input parameter.").

21. Referring to claim 19, Ko discloses the type and value information includes a range of values that are permitted for each of the multiple parameters (From line 57 of column 11, "A parameter is considered independent from another parameter if it may take on any value in its permitted range without affecting the constraints on the value of the other input parameter.").

22. Referring to claim 21, Ko discloses at least one of the multiple parameters is independent of type (From line 53 of column 11, "If the parameters a and b are independent, then the traversal number assigned to the directed edge is the maximum number of values that either a or b must be tested according to the test data set listed above. A parameter is considered independent from another parameter if it may take on any value in its permitted range without affecting the constraints on the value of the other input parameter. According to the test data set the parameter a must be tested with the values 0, 5 and 9 and the parameter b must be tested with the values 1 and 4. There is no indication in the test data set or in the simplified EFSM 5 of FIG. 1 that the parameters a and b are dependent on one another. Thus, the traversal number T.sub.1,111 is assigned a value 3, as indicated by a label 317 in FIG. 4, because it is

the maximum number either parameter a or b must be evaluated with during conformance testing. However, if the parameters a and b were not independent then the corresponding traversal number needs to be changed accordingly. In such an instance, the traversal number T.sub.1,11 must be assigned a value equal to the greater of the sum of the dependent parameter combinations or the maximum number any one of the parameters needs to be tested according to the test data sequence. For instance, if the parameters a and b were not independent upon each other then the directed edge 315 may have to be traversed with particular combinations of the parameters a and b.").

23. Referring to claim 22, Ko discloses a system that behaves in accordance with a state machine, comprising: a processor for implementing and using a state machine; a memory coupled to the processor, the memory storing information about the state machine; a bus; and a device coupled to the processor via the bus, the device capable of being tested by the state machine (Figure 2.),

wherein the state machine corresponds to a function that is stored in generic parameter independent code (From the abstract, with emphasis, "**A** method and apparatus for generating a conformance test data sequence of minimal length to verify that a device conforms to a protocol entity which can be characterized by a simplified extended finite state machine. **The** method generates an expanded directed graph of the protocol wherein each state is represented by a state vertex and a dummy vertex which are connected to other similar state and dummy vertices by directed edges in a configuration corresponding to the operation of the machine." See figure 3 where the test data set is retrieved following the steps of converting and generating.), wherein the state machine has a current state (It is a state machine.).

24. Referring to claim 23, Ko discloses the state machine is implemented as a linked list of states (See figures 1, 4-8, 10.).

25. Referring to claims 24-29, Ko discloses each of the states corresponds to a state of the state machine, wherein the state machine receives inputs that determines a next state and an output, a next state, a next next state, a previous state, and a previous previous state for each state (It is a state machine. See figures 1, 4-8, 10.).

26. **Claims 30-43 rejected under 35 U.S.C. 102(b) as being anticipated by US 6002869 to Hinckley.** Referring to claim 30, Hinckley discloses a method for varying parameters for a device under test, comprising: setting a current parameter to a first parameter (Figure 9, 908.);

setting a flag to indicate that testing is to continue (Figure 9, 918.);

and determining if the current parameter is to be randomized (Figure 9, 910. From line 6 of column 15, “If the test automation system 102 has been instructed to perform a random test...”).

27. Referring to claim 31, Hinckley discloses if it is determined that the current parameter is to be randomized, then randomizing the current parameter's value index (From line 8 of column 15, “then the test case generator 808 will randomly choose one of the possible next states. Referring to the example test file illustrated in FIG. 4, one of the Next States 404 defined in the state specification table 400 would be selected.”).

28. Referring to claim 32, Hinckley discloses determining if the current parameter is equal to the last parameter (From line 17 of column 15, "If the test automation system 102 has been instructed to perform a basis path test, the test case generator 808 starts from the first next state provided and verifies that all basis path tests have been performed for that next state. If there are more basis path tests to perform on that next state, then the test case generator 808 selects that next state. Otherwise, the test case generator 808 advances past that next state to and selects one of the following next states that has more basis path tests to perform." From line 11 of column 16, "This process is repeated for the additional basis path and range tests in the test specification file 204 as shown by block 918.").

29. Referring to claim 33, Hinckley discloses if it is determined that the current parameter is not the last parameter, then setting the current parameter to the next parameter (From line 17 of column 15, "If the test automation system 102 has been instructed to perform a basis path test, the test case generator 808 starts from the first next state provided and verifies that all basis path tests have been performed for that next state. If there are more basis path tests to perform on that next state, then the test case generator 808 selects that next state. Otherwise, the test case generator 808 advances past that next state to and selects one of the following next states that has more basis path tests to perform.").

30. Referring to claim 34, Hinckley discloses if the flag indicates that testing is to continue, determining if the current parameter's value is equal to the last parameter (From line 17 of column 15, "If the test automation system 102 has been instructed to

perform a basis path test, the test case generator 808 starts from the first next state provided and verifies that all basis path tests have been performed for that next state. If there are more basis path tests to perform on that next state, then the test case generator 808 selects that next state. Otherwise, the test case generator 808 advances past that next state to and selects one of the following next states that has more basis path tests to perform." From line 11 of column 16, "This process is repeated for the additional basis path and range tests in the test specification file 204 as shown by block 918.").

31. Referring to claim 35, Hinckley discloses if it is determined that the current parameter is not to be randomized, then determining if the current parameter's value is equal to the last parameter (From line 17 of column 15, "If the test automation system 102 has been instructed to perform a basis path test, the test case generator 808 starts from the first next state provided and verifies that all basis path tests have been performed for that next state. If there are more basis path tests to perform on that next state, then the test case generator 808 selects that next state. Otherwise, the test case generator 808 advances past that next state to and selects one of the following next states that has more basis path tests to perform." From line 11 of column 16, "This process is repeated for the additional basis path and range tests in the test specification file 204 as shown by block 918.").

32. Referring to claim 36, Hinckley discloses if it is determined that the current parameter is the last parameter, then checking the flag to indicate that testing is to continue (From line 17 of column 15, "If the test automation system 102 has been

instructed to perform a basis path test, the test case generator 808 starts from the first next state provided and verifies that all basis path tests have been performed for that next state. If there are more basis path tests to perform on that next state, then the test case generator 808 selects that next state. Otherwise, the test case generator 808 advances past that next state to and selects one of the following next states that has more basis path tests to perform." From line 11 of column 16, "This process is repeated for the additional basis path and range tests in the test specification file 204 as shown by block 918.").

33. Referring to claim 37, Hinckley discloses if checking the flag indicates that testing is to continue, testing the device under test with current parameter values (Figure 9, 918 then 916.).

34. Referring to claim 38, Hinckley discloses if checking the flag indicates that testing is not to continue, then stopping testing of the device under test (From line 13 of column 16, "Upon completion, testing ceases at stop block 920.").

35. Referring to claim 39, Hinckley discloses setting the current parameter to a new parameter for a new test cycle (Figure 3, 312, 316, 318, 304...).

36. Referring to claim 40, Hinckley discloses if it is determined that the current parameter is the last parameter, then setting the current parameter to a next parameter (Figure 9, 918. Figure 3, 312, 316, 318, 304...).

37. Referring to claim 41, Hinckley discloses determining if the current parameter is to be randomized (Figure 9, 910. From line 6 of column 15, "If the test automation system 102 has been instructed to perform a random test...").

38. Referring to claim 42, Hinckley discloses a function that uses the current parameter is implemented in common code (From the abstract (with emphasis), "**A** test automation system... **A** test engine...").

39. Referring to claim 43, Hinckley discloses the current parameter is independent of type (From line 13 of column 15, "Referring again to the exemplary test specification file 400, the test case generator 808 would, for example, set the minimum file cache buffers at line 38 of test specification 500 to a random value between 20 and 100." From line 25 of column 15, "Once the next state is determined, then the value for the associated test function 202 is selected. In one aspect of the invention, when there is an entry in the Min and Max fields, the test case generator 808 chooses a value between the minimum and maximum values to use for the test function." Wherein the latter does not specify a "type".).

Claim Rejections - 35 USC § 103

40. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

41. **Claim 7 rejected under 35 U.S.C. 103(a) as being unpatentable over US 5623499 to Ko et al. as applied to claim 1 above, and further in view of JP 10021286 A to Iwasaki.** Although Ko has not specifically disclosed the number of states needed to build the state machine may be estimated, doing so is known in the art. An example of this is shown by Iwasaki, "the number of gates of the state machine

is estimated". A person of ordinary skill in the art at the time of the invention would have been motivated to estimate, from Iwasaki, "to provide the circuit scale estimating method which precisely estimates the circuit scale in consideration of a logical function and common terms of a state machine by inputting a function description."

42. **Claim 14 rejected under 35 U.S.C. 103(a) as being unpatentable over US 5623499 to Ko et al. as applied to claim 12 above, and further in view of US 20030093608 to Jaramillo et al.** Referring to claim 14, although Ko does not specifically disclose the multiple parameters include a PCI cache line size, adjusting a PCI cache line size is known in the art. An example of this is shown by Jaramillo from paragraph 44, "This approach can be implemented by using other multiples or with a programmable multiple, or the standard PCI specification cache line size register can be adjusted such that the PCI to PCI bridge 350 actually prefetches multiple cache lines." A person of ordinary skill in the art at the time of the invention would have been motivated to use PCI cache line size as a parameter because, from paragraph 44 of Jaramillo, "It raises the overall system performance dramatically." Further, such a parameter would have been included for testing because Ko discloses from line 60 of column 15, "Further, although the example EFSM concerned a communications protocol, the present invention may be used for generating conformance test data sequences for any entity that can be characterized by a simplified EFSM." and Ko further discloses from line 10 of column 6, "The particular values of the input parameters and output parameters in the test data set may be specified by designers or conformance testers of the protocol entity. The test data set facilitates the exercise of a state transition several

times with different values of the same input and output parameter that may be required to verify that the operation of a device exactly conforms to the protocol entity."

43. **Claim 15 rejected under 35 U.S.C. 103(a) as being unpatentable over US 5623499 to Ko et al. as applied to claim 12 above, and further in view of US 6675244 to Elliot et al.** Referring to claim 15, although Ko does not specifically disclose the multiple parameters include a Small Computer System Interface (SCSI) synchronous rate, adjusting the SCSI system rate is known in the art. An example of this is shown by Elliot from line 34 of column 7, "The ASRT state 608 is a transitory state in which the timer is loaded with a value suitable for a delay discussed in conjunction with the next state, a WAIT_ASRT state 610. The value loaded into the timer during the ASRT state 608 depends on whether the linear mode is enabled, what the determined SCSI synchronous rate is, and whether this particular clock pulse is being "stretched". These aspects are further discussed below in conjunction with FIGS. 8-12. To summarize, if the linear mode is enabled, the SCSI clock will be asserted for a number of repeater 40 clock cycles that most closely matches the incoming clock signal from the other side of the repeater 40, but with some degree of "snapping" when the rate is near a standard SCSI rate." A person of ordinary skill in the art at the time of the invention would have been motivated to include a SCSI synchronous rate because, as disclosed by Elliot, the rate affects system performance. Further, such a parameter would have been included for testing because Ko discloses from line 60 of column 15, "Further, although the example EFSM concerned a communications protocol, the present invention may be used for generating conformance test data sequences for any

entity that can be characterized by a simplified EFSM." and Ko further discloses from line 10 of column 6, "The particular values of the input parameters and output parameters in the test data set may be specified by designers or conformance testers of the protocol entity. The test data set facilitates the exercise of a state transition several times with different values of the same input and output parameter that may be required to verify that the operation of a device exactly conforms to the protocol entity."

44. **Claim 16 rejected under 35 U.S.C. 103(a) as being unpatentable over US 5623499 to Ko et al. as applied to claim 12 above, and further in view of "block size" by Microsoft Computer Dictionary (MSCD).** Referring to claim 16, although Ko does not specifically disclose the multiple parameters include block size, adjusting the block size is known in the art. An example of this is shown by MSCD, "The declared size of a block of data transferred internally within a computer, via FTP, or by modem. The size is usually chosen to make most efficient use of all the hardware devices involved." A person of ordinary skill in the art at the time of the invention would have been motivated to include a block size because, as disclosed by MSCD, the block size affects system performance. Further, such a parameter would have been included for testing because Ko discloses from line 60 of column 15, "Further, although the example EFSM concerned a communications protocol, the present invention may be used for generating conformance test data sequences for any entity that can be characterized by a simplified EFSM." and Ko further discloses from line 10 of column 6, "The particular values of the input parameters and output parameters in the test data set may be specified by designers or conformance testers of the protocol entity. The test data set

facilitates the exercise of a state transition several times with different values of the same input and output parameter that may be required to verify that the operation of a device exactly conforms to the protocol entity."

45. **Claim 20 rejected under 35 U.S.C. 103(a) as being unpatentable over US 5623499 to Ko et al. as applied to claim 19 above, and further in view of US 6546507 to Coyle et al.** Referring to claim 20, Ko discloses that parameters in the test data set have a range that is stepped through (From line 57 of column 11, "A parameter is considered independent from another parameter if it may take on any value in its permitted range without affecting the constraints on the value of the other input parameter."). Although Ko does not specifically disclose the type and value information includes an incremental step size for each of the multiple parameters, using an incremental step for testing values is known in the art. An example of this is shown by Coyle, from line 46 of column 39, "The method 2850 starts at block 2852, where it verifies that the system operates correctly at a given initial value. The method 2850 in block 2854 tests whether the system passes at that value. If it does not, block 2856 reports a system failure. If the system passes, block 2858 adjusts the initial value to a new value, e.g., a single step up in the value. Block 2862 tests the system at this new value, and, if it passes, block 2862 saves the new value to a variable called HIGHGOOD. Then, method 2850 returns to block 2858 where the value can be again incremented in the same direction. If the test of block 2860 fails, the method 2850 proceeds to block 2864, where it resets the parameter to the initial value. Then, block 2866 tests whether the system passes at this value. If it does not, then block 2868

reports a system failure. If the system passes, block 2872 adjusts the parameter in the opposite direction to that of block 2858, e.g., a single step down. In other words, one of the blocks 2858 and 2872 increments the parameter value to test the operational limit in one direction, while the other decrements that value to test the operational limit in the other direction." A person of ordinary skill in the art at the time of the invention would have been motivated to incrementally step because as disclosed by Coyle, it permits operational envelope testing, and further as disclosed by Ko, there is a range of values that need to be tested for a given parameter.

Allowable Subject Matter

46. **Claim 13 objected to as being dependent upon a rejected base claim, and itself rejected under USC 101 above, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.** Referring to claim 13, the prior art does not teach or fairly suggest, in light of its parent claims, the at least one function includes at least one of the group consisting of editing, storing, loading, and displaying.

Conclusion

47. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. See notice of references cited.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Gabriel L. Chu whose telephone number is (571) 272-

3656. The examiner can normally be reached on weekdays between 8:30 AM and 5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Scott Baderman can be reached on (571) 272-3644. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Gabriel L. Chu
Examiner
Art Unit 2114

gc